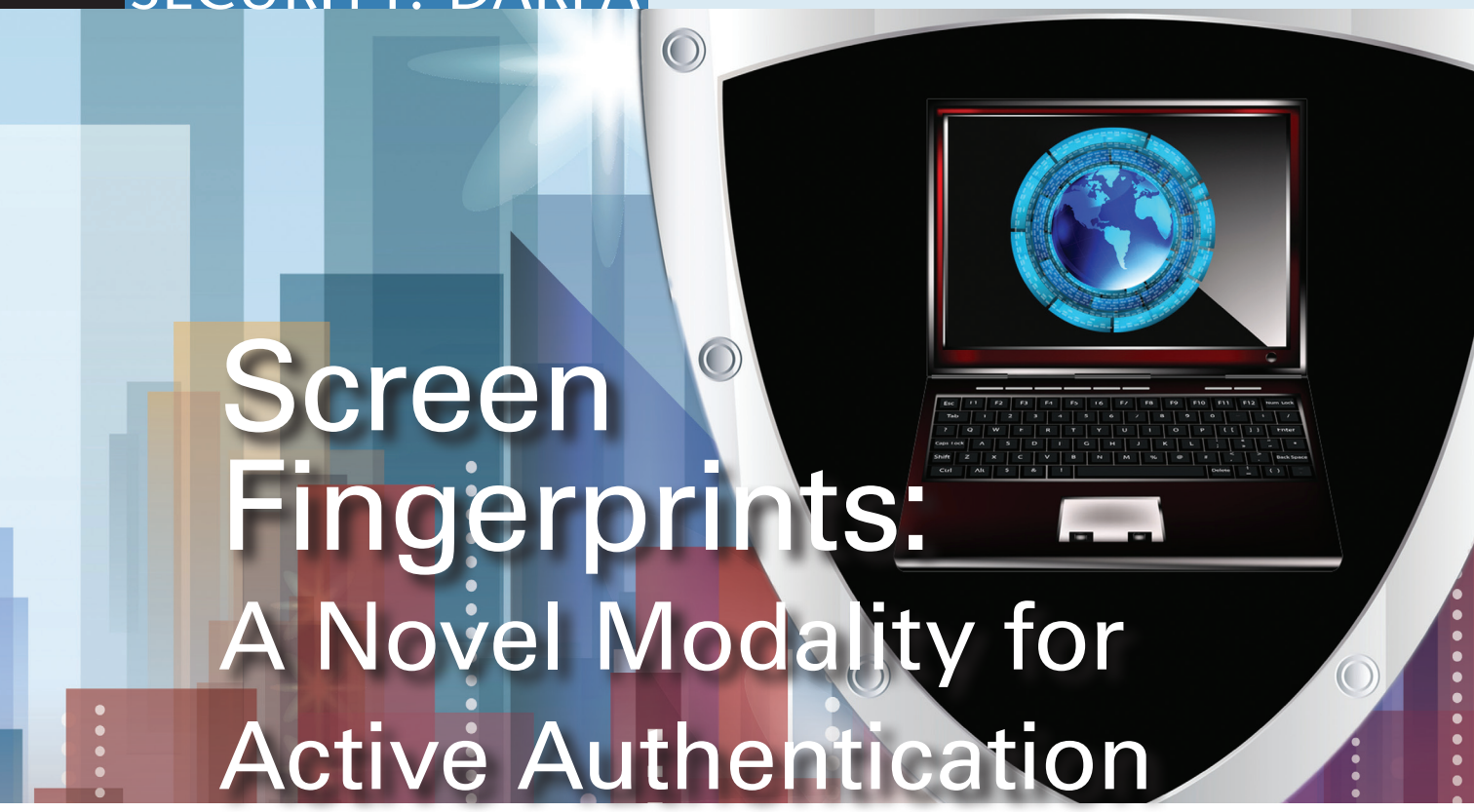

Screen Fingerprints: A Novel Modality for Active Authentication

Vishal M. Patel, *University of Maryland, College Park*
Tom Yeh, *University of Colorado, Boulder*
Mohammed E. Fathy and Yangmuzi Zhang, *University of Maryland, College Park*
Yan Chen, *University of Colorado, Boulder*
Rama Chellappa and Larry Davis, *University of Maryland, College Park*

A screen fingerprint is proposed as a new biometric modality for active authentication. Such a fingerprint is acquired by taking a screen recording of the computer being used and extracting discriminative visual features from the recording.

We propose a novel way of validating the identity of the person at a console by using a *screen fingerprint.* This new cyberbiometric can help measure and analyze active authentication. We acquire a screen fingerprint by taking a screen recording of the computer being used and extracting discriminative visual features from the recording.

The screen fingerprint of an operator captures enough unique human qualities for use as a biometric for authentication. The qualities captured include cognitive abilities, motor limitations, subjective preferences, and work patterns. For example, how well the operator sees is a cognitive ability that can be captured visually by the size of the text shown on the screen. How fast the operator drags a window is a motor limitation that can be captured visually by the amount of motion detected on the screen. How the operator arranges multiple windows is a preference that can be captured visually by the layout of salient edges identified on the screen. What suite of applications the operator uses is a work pattern that can be captured visually by the distribution of application-specific visual features recognized on the screen.

The proposed technology exploits the synergy between recent advances in pixel-level screen analysis[1–3] and vision-based biometrics, such as

 Published by the IEEE Computer Society 1520-9202/13/$31.00 © 2013 IEEE

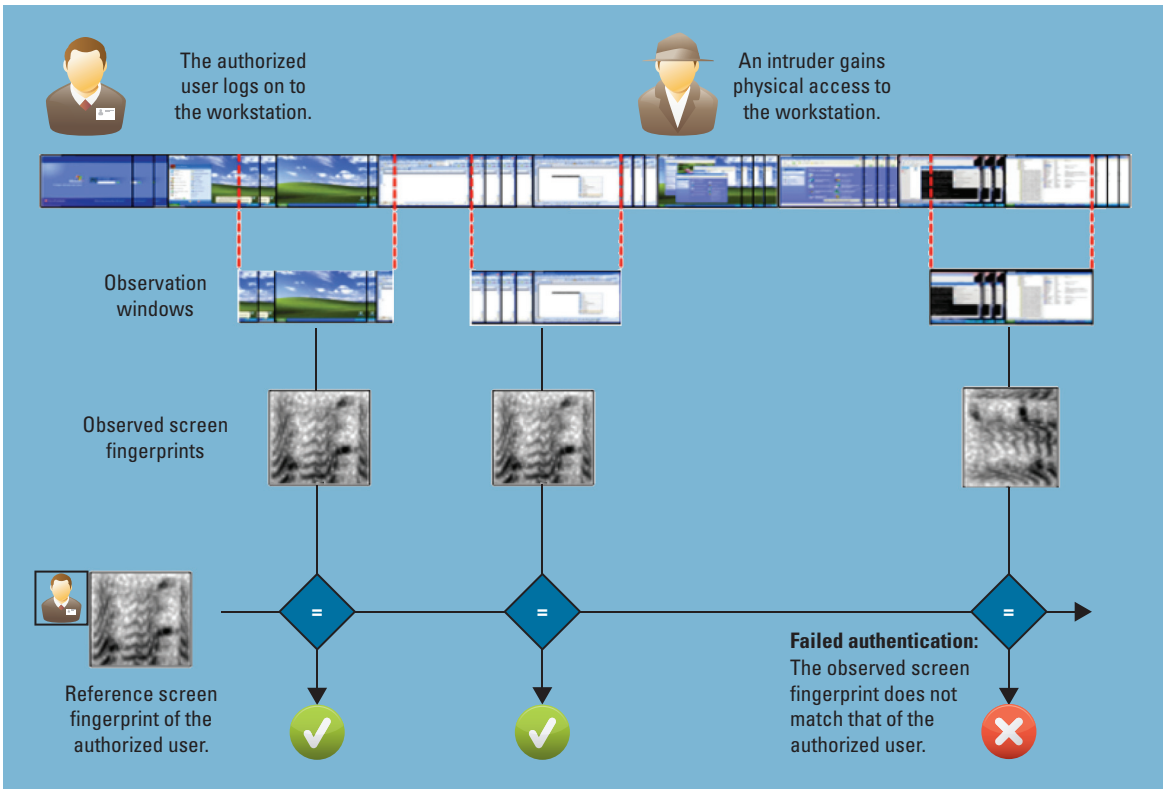

Figure 1. A typical scenario of active authentication using screen fingerprints.

face and iris recognition. Vision-based biometrics depend on hardware sensors that often limit its applicability. On the other hand, pixel-level screen analysis has received a lot of attention in human-computer interaction in the past two years, and it has the advantage of wide applicability because the screen buffer can be accessed on all platforms at the software level.

However, pixel-level screen analysis hasn't been used as a modality for biometrics. This is the first attempt to combine vision-based biometrics and pixel-level screen analysis in a complementary manner for active authentication. The scheme can also be regarded as a screen-based active intrusion detection technique, because it has the same two basic steps of intrusion detection: actively logging user activity (as seen on the screen) and analyzing it to detect possible attacks.[4]

Screen Fingerprints

For the past few years, researchers have applied computer vision techniques to the analysis of GUI screen recordings in support of a wide range of applications, including automation,[5] search,[3] software testing,[1] and tutorials.[2] Some applications perform batch analysis after screen recordings are acquired, such as searching online documentation about the interfaces in a screen recording.[2,3] Some applications operate in real time while the recordings are made. For example, Tom Yeh and his colleagues developed the Sikuli visual automation tool,[5] which can observe a screen recording in real time, identify an interface component by appearance, and send an automation command (such as "click") to that component. This tool has significantly affected software engineering in that dozens of companies use it to automate GUI testing.

Active authentication based on screen fingerprints is a novel screen-recording application (see Figure 1). First, an operator logs on the computer, using an initial authentication mechanism such as a password. While the operator is using the workstation, a program observes the computer screen and takes screen recordings within short observation windows. Each time a screen recording is taken, the recording (a video) is visually analyzed to extract a screen fingerprint to identify the person using the computer. This observed screen fingerprint is compared to the reference screen fingerprint, previously measured and stored for the authorized operator. If a match is established between the observed and reference fingerprints, the operator is actively authenticated.

Now suppose the operator steps away and leaves the workstation unattended. An adversary could gain physical access to the computer. While the adversary is using the computer, a screen recording is taken to extract a screen fingerprint. However, the observed screen fingerprint no longer matches the reference screen fingerprint, so active authentication fails. The workstation can lock itself to prevent further unauthorized use by the adversary.

The system is intended to be a component in a larger monitoring system. In case it detects an attack, it sends an alert to the monitoring system, which specifies what action to take—such as locking the computer and asking the user to provide a password and answer secret questions. If the user provides the extra credentials, the attack is flagged as false, and the corresponding set of features can be used to adapt the trained classifier.

Advantages

Screen fingerprints offer several advantages over other potential modalities for active authentication.

Relies on Visual Cues

Language-based techniques, such as those based on computational linguistic and structural semantic analysis, seek to authenticate computer operators on the basis of verbal cues, such as the words and phrases an operator uses in digital communication (emails or memos, for example). These stylometry techniques[6,7] don't work well in situations, such as data entry, where operators' primary responsibilities don't involve personal communication or when operators mainly use mouse or touch-based interfaces, such as with Photoshop. Our proposed modality can deal with these situations, because it relies on visual cues that are always observable on a computer screen regardless of the types of applications operators use.

Supports Voice and Touch Modalities

Motor-based techniques seek to authenticate computer operators based on kinetic cues, such as how fast an operator types or moves a mouse pointer. These techniques can't support operators who use voice or touch as the primary input modality. Our proposed modality can authenticate operators who don't use a mouse or keyboard, because it doesn't depend on specific input devices.

Offers Comprehensive Coverage

Application-based techniques seek to authenticate computer operators according to usage cues, such as which applications or features an operator is using. However, these techniques are difficult to scale, because each application must be specifically instrumented to track its usage. Often, such deep instrumentation requires access to the application's source code or a special API. Comprehensive coverage is difficult to attain, because some proprietary or legacy applications don't provide source code or the API for instrumentation.

Our proposed modality provides wide coverage over most applications without deep instrumentation. As long as an application is visible on a computer screen, it can be captured in a screen recording. Some of an application's distinctive visual properties can be extracted to be part of an operator's screen fingerprint.

Experiments

To study the effectiveness of screen fingerprints for active authentication, we created a dataset of screen recordings taken under different scenarios. The first dataset contains screen recordings of 21 users. We asked each user to do a simple task—such as dragging and dropping a file, scrolling a PDF document, typing a paragraph, or resizing a window—five times. It takes 15 to 25 minutes to collect data from a single user. A Java program collected the data by capturing screen recordings at a rate of 12 frames per second. The program kept each user's data in an independent directory and called batch files to set the stage for and start each task.

To determine whether the screen recordings were discriminative of the different individuals, we calculated the pixel change between each pair of consecutive frames of the screen fingerprint. We then calculated the histogram of the pixel change and used the extracted information as features. The data for eight individuals data was projected onto the first two eigenvectors, determined by the pixel-change features. Figure 2 shows the 2D data, indicating that different individuals perform the same task differently.

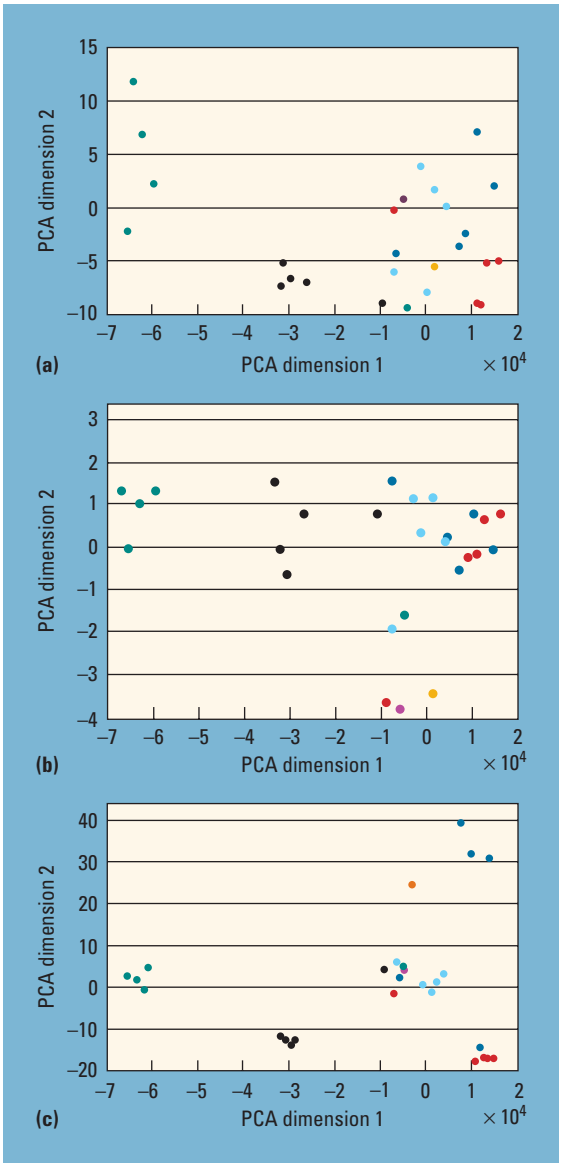

Figure 2. Screen fingerprints capture the unique differences present in a task performed by different individuals: (a) drag and drop, (b) resizing, and (c) scrolling. The different colors represent different individuals (there are five samples for each person).

In the second set of experiments, we calculated from each sample recording the *average histogram of optical flow* as a descriptor. After that, for each interaction, we calculated the average false acceptance rate (FAR) and false rejection rate (FRR) by training a linear soft-margin support vector machine to verify each user against other users, using 80 percent of the samples of that interaction. The remaining 20 percent subset was used to test the classifier and calculate the FAR and FRR for that user. We repeated that process five times, and each time we tested on a different 20 percent subset of the data. Table 1 presents the average interaction time and the averages of the error rates from different users.

Table 1. The average false acceptance rate (FAR) and false rejection rate (FRR) for each interaction.

Interaction	Average time (sec)	Average FAR (%)	Average FRR (%)
Typing	43	37.57	36.19
Mouse moving	12	30.29	29.89
Scrolling	83	20.67	12.38
Other	2	21.85	33.82

The performance results obtained in our experiments might not be as high as those for other well-established modalities, such as mouse dynamics. However, screen output can enhance the security of a multimodal system when there's little data from other monitored modalities. Furthermore, the performance of the other modalities is the result of many years of research (33 years for keystroke dynamics[8] and nine years for mouse dynamics[9]), so further research into screen fingerprints, investigating richer features and other classifiers, should improve the performance.

Acknowledgments

This work was supported by cooperative agreement FA87501220199 from DARPA.

References

1. T.-H. Chang, T. Yeh, and R. Miller, "GUI Testing Using Computer Vision," *Proc. Conf. Human Factors in Computing System*, ACM, 2010, pp. 1535–1544.
2. T. Yeh et al., "Creating Contextual Help for GUIs Using Screenshots," *Proc. 24th ACM Symp. User Interface Software and Technology*, ACM, 2011, pp. 145–154.
3. T. Yeh et al., "A Case for Query by Image and Text Content: Searching Computer Help Using Screenshots and Keywords," *Proc. 20th Int'l Conf. World Wide Web*, ACM, 2011, pp. 775–784.
4. R. Sandhu and P. Samarati, "Authentication, Access Control and Intrusion Detection," *CRC Handbook of Computer Science and Engineering*, CRC Press, 1997, pp. 40–48.
5. T. Yeh, T.-H. Chang, and R. Miller, "Sikuli: Using GUI Screenshots for Search and Automation," *Proc. 22nd Ann. ACM Symp. User Interface Software and Technology*, 2009, ACM, pp. 183–192.

6. K. Calix et al., "Stylometry for E-Mail Author Identification and Authentication," *Proc. Student-Faculty CSIS Research Day*, Pace University, 2008.
7. O. Canales et al., "A Stylometry System for Authenticating Students Taking Online Tests," *Proc. Student-Faculty CSIS Research Day,* Pace University, 2011.
8. R.S. Gaines et al., *Authentication by Keystroke Timing: Some Preliminary Results*, tech. report, Defense Technical Information Center (DTIC) Document, 1980.
9. M.S.S. Hashia and C. Pollett, "On Using Mouse Movements as a Biometric," *Proc. Int'l Conf. Computer Science and its Applications*, ACM, 2004, pp. 1–8.

Vishal M. Patel *is a member of the research faculty at the University of Maryland, College Park. Contact him at pvishalm@umiacs.umd.edu.*

Tom Yeh *is an assistant professor in the Department of Computer Science at the University of Colorado, Boulder. Contact him at tom.yeh@colorado.edu.*

Mohammed E. Fathy *is a PhD student in the computer science program at the University of Maryland, College Park. Contact him at mefathy@umiacs.umd.edu.*

Yangmuzi Zhang *is PhD student in electrical and computer engineering at the University of Maryland, College Park. Contact him at ymzhang@umiacs.umd.edu.*

Yan Chen *is pursuing a double major in Applied Math and Electrical Engineering at the University of Colorado, Boulder. Contact him at yan.chen@colorado.edu.*

Rama Chellappa *is a professor of electrical engineering and an affiliate professor of computer science at the University of Maryland, College Park. Contact him at rama@umiacs.umd.edu.*

Larry Davis *is a professor in the Department of Computer Science and at the Center for Automation Research at the University of Maryland, College Park. Contact him at lsd@umiacs.umd.edu.*

Selected CS articles and columns are available for free at http://ComputingNow.computer.org.